JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p.5514-5520
https://publishoa.com
ISSN: 1309-3452

An Efficient Recurrent Neural Network based Classification Method for Cyber Threat Detection Analysis

T.Elangovan[*]
[*]Ph.D. Research Scholar, Dept. of Computer Science Erode Arts and Science College,
Erode-638 009, Tamilnadu, India. E-mail: elangovaneasc@gmail.com

Dr.S.Sukumaran
Associate Professor in Computer Science, Erode Arts and Science College,
Erode-638 009, Tamilnadu, India. E-mail: prof_sukumar@yahoo.co.in

Dr.S.Muthumarilakshmi
Associate Professor, S.A. Engineering College,
Chennai-600 077, Tamilnadu, India. E-mail: muthumarilakshmi827@gmail.com

Accepted: 2022 May; November, 16; **Published**: 2022 December, 24

Abstract

Cyber threat detection plays an important role in ensuring information security, and the key technology is to accurately identify various attacks in the network. Security of the computer systems is the most important factor for single users and businesses, because an attack on a system can cause data loss and considerable harm to the business. Due to the increment of the range of the cyber-attacks, antivirus scanners cannot fulfill the need for protection. Hence, the increment of the skill level that required for the development of cyber threats and the availability of the attacking tools on the internet, the need for Artificial Intelligence based systems, is a must to the users. In this study, an intrusion detection system based on deep learning, and proposes a deep learning approach for intrusion detection using Efficient Recurrent Neural Networks (ERNN). Moreover, the performance of the model in binary classification and multiclass classification, and the number of neurons and different learning rate impacts on the performance of the proposed model. We compare it with those of Recurrent Neural Network and Support Vector Machine proposed by previous researchers on the benchmark dataset. The experimental results show that ERNN intrusion detection system is very suitable for modeling a classification method with high accuracy and that its performance is superior to that of traditional machine learning classification methods. The ERNN cyber threat detection method improves the accuracy of the intrusion detection and provides a new research method for cyber threat detection.

Index Terms — Cyber Threat Detection, Efficient Recurrent Neural Networks, Intrusion Detection, Deep Learning, Machine Learning.

1. Introduction

Cyber-attacks divided according to the layer of the network infrastructure, which is targeted by an attacker. They can be categorized as User Threats, Application Threats, and Infrastructure Threats [1]. Attackers deploy the vulnerabilities to the specific layer and design them to change their properties according to the environment. There are four methods of detecting vulnerabilities. Signature based detection methods, Static analysis, Dynamic analysis and Heuristics based analysis. There are several systems and applications developed using these detection methods. They are created to work in different environments [2]. But the findings of some observations prove that detecting systems become vulnerable and occur several problems when they are running. This paper properly explains the ways to minimize errors with the use of Artificial Intelligence and Machine Learning.

The cyberspace refers to the global environment that facilitates the sharing of electronic resources from all over the world. Resources can be an electronic document, audio, video, image, and tweet [3]. The cyberspace incorporates a wide range of components, including the Internet, technically skilled users, system resources, data and untrained users. The cyberspace is providing a global arena to infinitely gain access to information and resources. At present, the cyberspace is playing the leading role in data transfer and information exchange with all its vastly growing losses and gains. The elevating cyberspace has also given rise to the risks of cybercrimes and cyber threats. With the growing range of cyber threats, cyber security has also made a considerable number of enhancements to compete against cybercrimes. The cyber security refers to a set of technologies, technology experts and processes that are used to make safety measures to protect the cyberspace from cybercriminals [4].

The cyber threat is an act in which someone will try or attend to steal the information, violate the integrity rules and harm the computing device or network. Cyber threats include phishing, malware, attack on IoT devices, denial of service attack, spam, intrusion on network or mobile device, financial fraud, ransom ware, to name a few [5, 6]. An email that is unwanted or unsolicited is called spam email. Spam emails are mostly used for advertisement or spreading fraudulent material. It occupies the network and computer resources such as the bandwidth of network, memory and wastage of time [7]. Another cyber threat is malware. Malware, as short for malicious software, is a software that is installed on a computer to disrupt its operation and harm the electronic data. Viruses, worms, ransom ware, adware, spyware, malvertising, and Trojan horse are considered as significant types of malware [8]. Malign intrusions over the computer network and devices are another cyber threat to cyberspace. These intrusions are used to identify and scan the vulnerabilities of a network or computer system.

An intrusion detection system (IDS) is used to protect against these intrusions. Machine learning (ML) is the most effective and fundamental strategy to compete against cyber threats and overcome the limitations of conventional security systems [11]. Despite having all its charms, machine learning techniques have their constraints and limitations. The fascinating quality of machine learning techniques is that machine learning techniques do not need to be explicitly programmed as they can automatically learn from their experience to generate the results [13]. On the strength of all the benefits of machine learning techniques, ML techniques are expanding their scope in almost every area of life, including cyber security, medical science, educational purposes, intrusion detection, spam detection and malware detection. Almost all famous machine learning techniques have been applied to detect and classify different cyber threats.

2. Literature Review

V. Ford et. al., analyzed the applications of widely used machine learning techniques to protect the cyberspace from cybercriminals [6]. The authors also depicted various obstacles faced during the implementation of machine learning techniques. The work concluded that although the machine learning techniques are expanding various ways to protect cyberspace against cybercriminals, still there is an immense number of advancements needed to protect the classifiers from adversarial attacks. Machine learning classifiers themselves are incredibly vulnerable to cyber threats and adversarial attacks.

H. Jiang et. al., bestowed a brief review of several publications related to the implementation of machine learning models to enhance cyber security [9]. They addressed some commonly faced barriers to machine learning techniques in finding appropriate datasets with most efficient applicability for a specific security problem.

S. Sheikhi et. al., proposed a novel machine learning technique for spam detection in text messages using content based features [16]. They concluded that the proposed averaged neural network and content-based feature selection outplayed most of the recent machine learning techniques in terms of accuracy on the same dataset.

F. Mercaldo et. al., stated that the signature-based classification techniques generate results with high error rates when it comes to mobile malware detection [12]. They proposed an image-based deep learning technique for mobile malware detection, aiming to demonstrate the discrimination between the family of malicious attributes and the legitimate attributes by obtaining grey-scale images.

E. Hodo et. al., presented a brief performance comparison of different machine learning techniques, specifically in anomaly detection [7]. They gauged the performance efficiency of feature selection in ML for IDS. They claimed that the convolutional neural network (CNN) classifier is an underused classifier and it could have brought vast advancements in cyber security if it was used to its full potential.

G. Apruzzese et. al., analyzed the role of various machine learning techniques in spam detection, malware detection and intrusion detection [1]. They claimed that there is no machine learning technique that is not vulnerable to cyber-attacks. Every machine learning technique is still struggling to keep a pace with continuously upgrading cybercrimes.

Shadi Aljawarneh et. al., proposed a model that can effectively gather the sensitive information from the data [15]. Meta heuristic data is taken in this model and assessments are performed on the meta data that are used for identifying the possibility of the intrusion in the data. NSL KDD which is one of the famous dataset for intrusion detection was used. This is a hybrid model that can also identify the degrees up to which the intrusion has occurred in the network.

J. Ribeiro et. al., came up with a statistical semi-supervised machine learning technique for intrusion detection in Android mobile devices [14]. The increase in data traffic will also give rise to cybercrimes. Consequently, to protect Android mobile devices against advanced cybercrimes, more advanced machine learning techniques are needed to be developed to detect malicious activities.

3. Simple Recurrent Neural Network

Simple RNN also called Elman network, represents a fully connected network with a feedback [5]. The loop keeps the hidden state vector at a previous time step h(t-1) and feeds it with the new input vector x(t). Therefore, Simple RNN has the most general topology and most similar to the regular neural networks architectures.

The initial value (usually set to 0) of the hidden state vector is denoted by h(0). Hidden state vector h(t) at a time step t is calculated as follows:

$$h(t) = \sigma(W_{xh}x(t) + W_{hh}h(t-1) + b_h \qquad \dots (1)$$

where

b_h is a bias vector

σ represents the activation function

W_{xh} and W_{hh} denote the input and hidden weight matrices

RNN output y(t) is defined by the following equation:

$$y(t) = \sigma\ (W_{\mathbf{ho}}h(t) + b_o) \qquad \dots (2)$$

where σ is output activation function

W_{ho} represents output weight matrix

b_o is a bias vector

4. Proposed Methodology

4.1 Data Collection

Data collection is the first and a critical step to cyber threat detection. The type of data source and the location where data is collected from are two determinate factors in the design and the effectiveness of an IDS. To provide the best suited protection for the targeted host or networks, this study proposes a network-based IDS to test our proposed approaches. The proposed IDS runs on the nearest router to the victim and monitors the inbound network traffic. During the training stage, the collected data samples are categorized with respect to the transport/Internet layer protocols and are labeled against the domain knowledge. However, the data collected in the test stage are categorized according to the protocol types only.

4.2 Data Preprocessing

The data obtained during the phase of data collection are first processed to generate the basic features such as the ones in NSLKDD dataset. Data Preprocessing contains three main stages shown as follows.

4.2.1 Data Transferring

The trained classifier requires each record in the input data to be represented as a vector of real number. Thus, every symbolic feature in a dataset is first converted into a numerical value. For example, the NSLKDD dataset contains numerical as well as symbolic features. These symbolic features include the type of protocol (TCP, UDP and ICMP), service type (HTTP, FTP, Telnet and so on) and TCP status flag (SF, REJ and so on). The method simply replaces the values of the categorical attributes with numeric values.

4.2.2 Data Normalization

An essential step of data preprocessing after transferring all symbolic attributes into numerical values is normalization. Data normalization is a process of scaling the value of each attribute into a well-proportioned range, so that the bias in favor of features with greater values is eliminated from the dataset. Every feature within each record is normalized by the respective maximum value and falls into the same range of [0-1]. The transferring and normalization process will also be applied to test data. For NSLKDD and to make a comparison with those systems that have been evaluated on different types of attacks we construct five classes. One of these classes contains purely the normal records and the other four hold different types of attacks (i.e., DoS, Probe, U2R, R2L), respectively.

$$x_{i2} = \frac{x_{i1} - x_{min}}{x_{max} - x_{min}} \qquad \ldots (3)$$

4.2.3 Feature Selection

Even though every connection in a dataset is represented by various features, not all of these features are needed to build cyber threat detection. Therefore, it is important to identify the most informative features of traffic data to achieve higher performance. The proposed feature selection algorithms can only rank features in terms of their relevance but they cannot reveal the best number of features that are needed to train a classifier. Therefore, this study applies the same technique proposed in to determine the optimal number of required features. The technique first utilizes the proposed feature selection algorithm to rank all features based on their importance to the classification processes. Then, incrementally the technique adds features to the classifier one by one. The final decision of the optimal number of features in each method is taken once the highest classification accuracy in the dataset is achieved. In addition, NSLKDD, the proposed ERNN feature selection algorithm is applied for the aforementioned classes.

4.3 An Efficient Recurrent Neural Network Classification Method

Recurrent neural networks include input units, output units and hidden units, and the hidden unit completes the most important work. The RNN model essentially has a one-way flow of information from the input units to the hidden units, and the synthesis of the one-way information flow from the previous temporal concealment unit to the current timing hiding unit is shown in Fig. 1. We can regard hidden units as the storage of the whole network, which remember the end-to-end information. When we unfold the RNN, we can find that it embodies the deep learning. A RNNs approach can be used for supervised classification learning.

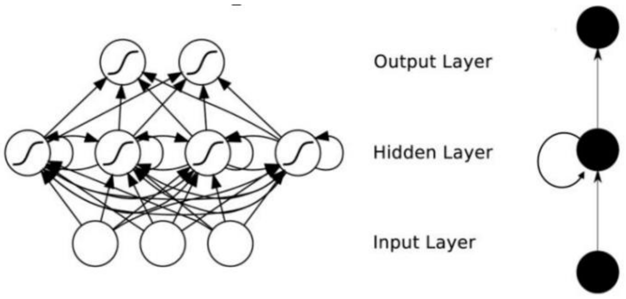

Fig 1. Recurrent neural networks

Recurrent neural networks have introduced a directional loop that can memorize the previous information and apply it to the current output, which is the essential difference from traditional Feed-forward Neural Networks (FNNs). The preceding output is also related to the current output of a sequence, and the nodes between the hidden layers are no longer connectionless; instead, they have connections. Not only the output of the input layer but also the output of the last hidden layer acts on the input of the hidden layer.

It is obvious that the training of the RNN-IDS model consists of two parts - Forward Propagation and Back Propagation. Forward Propagation is responsible for calculating the output values, and Back Propagation is responsible for passing the residuals that were accumulated to update the weights, which is not fundamentally different from the normal neural network training.

According to Fig. 1, an Unfolded recurrent neural network is presented in Fig. 2. The standard RNN is formalized as follows: Given training samples x_i (i = 1, 2, ..., m), a sequence of hidden states h_i (i = 1, 2, ..., m), and a sequence of predictions $\hat{y}_i$ (i = 1, 2, ..., m). W_{hx} is the input-to-hidden weight matrix, W_{hh} is the hidden-to-hidden weight matrix, W_{yh} is the hidden-to-output weight matrix, and the vectors b_h and b_y are the biases [26]. The activation function *e* is a sigmoid, and the classification function *g* engages the *SoftMax* function.

$$t_i = w_{hx}x_i + w_{hh}h_{i-1} + b_h \qquad \ldots (4)$$

$$h_i = sigmoid\ (t_i) \qquad \ldots (5)$$

$$s_i = w_{yh}h_i + b_y \quad \dots (6)$$

$$\hat{y}_i = SoftMax(s_i) \quad \dots (7)$$

The objective function associated with RNNs for a single training pair (x_i, y_i) is defined as f(θ) =L(y_i:$\hat{y}_i$) [26], where L is a distance function which measures the deviation of the predictions ŷi from the actual labels y_i. Let η be the learning rate and k be the number of current iterations. Given a sequence of labels y_i (i = 1, 2, ..., m).

Calculate the cross entropy between the output value and label value:

$$L(y_i: \hat{y}_i) \leftarrow -\sum_i \quad \sum_i \quad y_{ij}\, log\, log\, (\hat{y}_{ij}) + (1 - y_{ij}) log\, (1 - \hat{y}_{ij}) \quad \dots (8)$$

Compute the partial derivative with respect to θ_i:

$$\delta_i \leftarrow dL/d\theta_i \quad \dots (9)$$

Update the Weight

$$\theta_i \leftarrow \theta_i + \eta\delta_i \quad \dots (10)$$

Suppose there is a deeper network with one input layer, three hidden layers and one output layer. Then like other neural networks, each hidden layer will have its own set of weights and biases, let's say, for hidden layer 1 the weights and biases are (w1, b1), (w2, b2) for second hidden layer and (w3, b3) for third hidden layer. This means that each of these layers are independent of each other, i.e. they do not memorize the previous outputs.

Algorithm

Step 1: Start the Process
Step 2: Select the NSLKDD Network Dataset
Step 3: Perform Data preprocessing
Step 4: Transfer symbolic features into numerical value
Step 5: Apply min-max Normalization
Step 6: A single time step of the input is provided to the network.
Step 7: Then calculate its current state using set of current input and the previous state.
Step 8: The current ht becomes ht-1 for the next time step.
Step 9: One can go as many time steps according to the problem and join the information from all the previous states.
Step 10: Once all the time steps are completed the final current state is used to calculate the output.
Step 11: The output is then compared to the actual output i.e the target output and the error is generated.
Step 12: The error is then back-propagated to the network to update the weights and hence the network (ERNN) is trained.
Step 13: Finally Classified Data

5. Evaluation Metrics

The Network Security Laboratory - Knowledge Discovery in Databases (NSL-KDD) data set is an improved version of the KDD'99 data set. It is a minor data set that delivers best evaluation of classifiers since redundant records are removed. Redundant records cause learning classifiers to be biased toward the more frequent records during training, as well as increasing classification accuracy whenever these same records appear in the test set. The testing set KDDTest^{+} contains 22,544 records.

The most important performance indicator Accuracy of cyber threat detection is used to measure the performance of the ERNN model.

Recall measures the ratio of correct classification by missed entries

$$Recall = \frac{TP}{TP + FN} \quad \dots (11)$$

Precision is the ratio of correct classifications to the incorrect classifications

$$Precision = \frac{TP}{TP + FP} \quad \dots (12)$$

Accuracy is the percentage of correct classifications

$$Accuracy = \frac{TP + TN}{TP + FN + FP + TN} \quad \dots (13)$$

F-Measure is the harmonic mean of precision and recall.

$$F - Measure = 2 * \frac{Precison \cdot Recall}{Precision + Recall} \quad \ldots(14)$$

In order to compare the performance of ERNN with the reduced-size RNN method proposed in [20], we constructed the testing set from NSLKDD dataset. Table 1 shows the confusion matrix of the ERNN on the test set NSLKDD in the five-category classification experiments.

Table 1. Confusion Matrix for the Five-Category Experiments on NSLKDD

Predicted Class Actual Class/	Normal	DoS	R2L	U2R	Probe
Normal	**9377**	88	2	6	238
DoS	1011	**6227**	125	0	95
R2L	2058	0	**680**	6	10
U2R	149	0	11	**23**	17
Probe	231	166	5	0	**2019**

In this experiment, the detection rate of the ERNN method gets 96.09% on the dataset, also higher than 94.85% in the existing method.

Table 2. Results of the Evaluation Metrics for the NSLKDD Dataset

Methods/Parameters	Accuracy	Precision	Recall	F-Measure
SVM	92.22	93.58	94.19	93.25
RNN	94.85	96.33	95.04	94.17
ERNN	96.09	98.36	98.12	96.26

6. Conclusion

The proposed ERNN method not only has a strong modeling ability for intrusion detection, but also has high accuracy in both binary and multiclass classification. Compared with traditional classification methods, such as SVM and RNN, the performance obtains a higher accuracy rate and detection rate with a low false positive rate, especially under the task of multiclass classification on the NSLKDD dataset. The proposed work can effectively improve both the accuracy of intrusion detection and the ability to recognize the intrusion type. The NSLKDD dataset is used to compare the evaluation results in terms of recall, precision, F-Measure, and accuracy. Different learning models are being used for specific different cyber threats. The experiments findings showed that it is possible to train the ERNN so that it can accurately identify normal as well as abnormal flows along with attack types of unknown attacks. For examining the constrained RNN performance, it compared to well-known method which is the best detector. From the experiments, the proposed ERNN is able to detect the anomaly accurately, especially when large-scale training data are used. The results suggest that the ERNN is able to accurately identify the cyber threats.

Reference

[1] G. Apruzzese, M. Colajanni, L. Ferretti, A. Guido, and M. Marchetti, "On the Effectiveness of Machine and Deep Learning for Cyber Security", 10th International Conference on Cyber Conflict (CyCon), IEEE, Pp. 371-390, 2018.

[2] Y. Chang, W. Li and Z. Yang, "Network Intrusion Detection Based on Random Forest and Support Vector Machine", IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC), Guangzhou, Pp. 635-638, 2017.

[3] C. Chen et al., "A Performance Evaluation of Machine Learning-Based Streaming Spam Tweets Detection", IEEE Transactions on Computational Social Systems, Vol. 2, No. 3, Pp. 65-76, 2015.

[4] A. Damodaran, F. Di Troia, C. A. Visaggio, T. H. Austin, and M. Stamp, "A Comparison of Static, Dynamic, and Hybrid Analysis for Malware Detection," Journal of Computer Virology and Hacking Techniques, Vol. 13, No. 1, Pp. 1-12, 2017.

[5] Dusan Nedeljkovic and Zivana Jakovljevic, “Cyber-attack detection method based on RNN”, 7th International Conference on Electrical, Electronic and Computing Engineering (IcETRAN 2020), at Belgrade, 2020.

[6] V. Ford and A. Siraj, "Applications of Machine Learning in Cyber Security", In Proceedings of the 27th International Conference on Computer Applications in Industry and Engineering, 2014.

[7] E. Hodo, X. Bellekens, A. Hamilton, C. Tachtatzis, and R. Atkinson, "Shallow and Deep Networks Intrusion Detection System: A Taxonomy and Survey", arXiv preprint arXiv:1701.02145, 2017.

[8] Q. Jamil and M. A. Shah, "Analysis of Machine Learning Solutions to Detect Malware In Android," In Sixth International Conference on Innovative Computing Technology (INTECH), IEEE, Pp. 226-232, 2016.

[9] H. Jiang, J. Nagra, and P. Ahammad, "Sok: Applying Machine Learning in Security-A Survey", arXiv preprint arXiv: 1611.03186, 2016.

[10] G. Karatas and O. K. Sahingoz, "Neural Network Based Intrusion Detection Systems with Different Training Functions", 6th International Symposium on Digital Forensic and Security (ISDFS), Antalya, Pp. 1-6, 2018.

[11] H. Kim, T. Cho, G.-J. Ahn, and J. H. Yi, "Risk Assessment of Mobile Applications Based on Machine Learned Malware Dataset", Multimedia Tools and Applications, Vol. 77, No. 4, Pp. 5027-5042, 2018.

[12] F. Mercaldo and A. Santone, "Deep Learning for Image-Based Mobile Malware Detection", Journal of Computer Virology and Hacking Techniques, Pp. 1-15, 2020.

[13] D. Moon, H. Im, I. Kim, and J. H. Park, "DTB-IDS: An Intrusion Detection System based on Decision Tree using Behavior Analysis for Preventing APT Attacks," The Journal of Supercomputing, Vol. 73, No. 7, Pp. 2881-2895, 2017.

[14] J. Ribeiro, F. B. Saghezchi, G. Mantas, J. Rodriguez, S. J. Shepherd, and R. A. Abd-Alhameed, "An Autonomous Host-Based Intrusion Detection System for Android Mobile Devices", Mobile Networks and Applications, Vol. 25, No. 1, Pp. 164-172, 2020.

[15] Shadi Aljawarneh, Monther Aldwairi, Muneer Bani Yassein, “Anomaly-based Intrusion Detection System through Feature Selection Analysis and Building Hybrid Efficient Model”, Journal of Computational Science, Vol. 25, Pp. 152-160, 2018.

[16] S. Sheikhi, M. Kheirabadi, and A. Bazzazi, "An Effective Model for SMS Spam Detection Using Content-based Features and Averaged Neural Network," International Journal of Engineering, Vol. 33, No. 2, Pp. 221-228, 2020.

[17] R. Vinayakumar, M. Alazab, K. Soman, P. Poornachandran, and S. Venkatraman, "Robust Intelligent Malware Detection using Deep Learning," IEEE Access, Vol. 7, Pp. 46717-46738, 2019.

[18] C. Yin, Y. Zhu, J. Fei and X. He, "A Deep Learning Approach for Intrusion Detection Using Recurrent Neural Networks," IEEE Access, Vol. 5, Pp. 21954-21961, 2017.